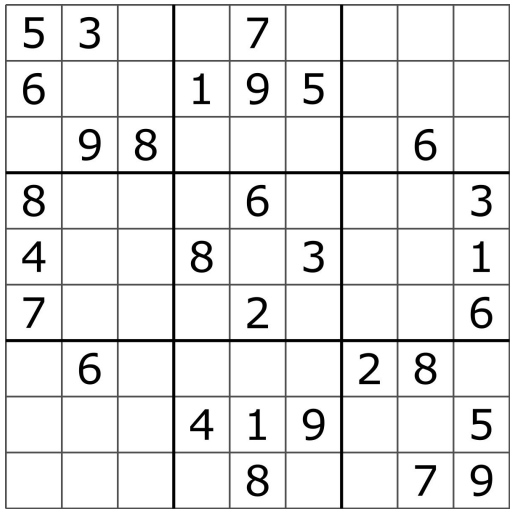
Parallelising Inference on Probabilistic Graphical Models

Author: Cornel Verburgh
Supervisors: Prof. Johan du Preez, Mr. Arno Barnard

What are PGMs used for?

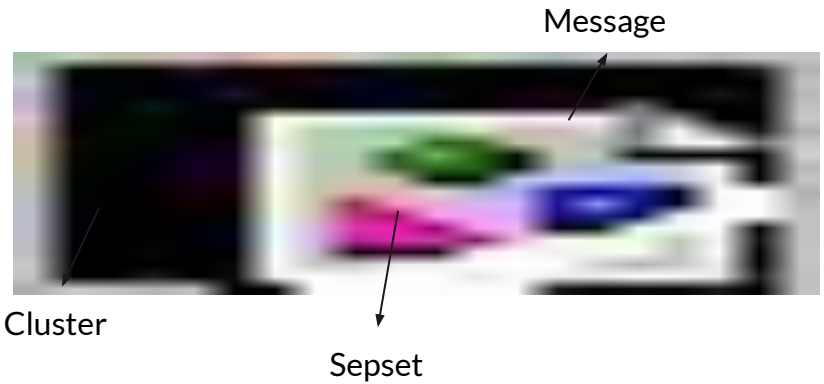


5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

- Topic Modelling
- Denoising Images
- Medical Diagnosis
- Solving Sudoku Puzzles
- Etc.

Probabilistic Graphical Models (PGMs)

Simple Cluster Graph:



Many representations of PGMs:

- Markov Random Fields
- Belief Networks
- Factor Graphs
- Cluster Graphs

Our focus:

- **Cluster Graphs**

Cluster Graph Explained

Clusters:

- Each cluster contains one or more factors.
- Each contain only one in this image.

Message Passing:

- m12 and m21 are messages.
- Messages are marginals over all variables not included in the sepsets (separation sets).

Inference

The distance between m12 and m12’ is calculated. When this distance is zero, the message has converged.

Passing messages until they converge is the process of **inference**.

In more complex PGMs, the order in which you pass these messages will determine the speed at which it converges, due to some messages carrying bigger changes compared to others.

Residual Belief Update (RBU)

This is how messages are scheduled for an update:

- A CPU thread pops a message from the queue.

Residual Belief Update (RBU)

This is how messages are scheduled for an update:

- A CPU thread pops a message from the queue.
- It calculates the new message.
- Then the distance between the old and new message is calculated.

Residual Belief Update (RBU)

This is how messages are scheduled for an update:

- A CPU thread pops a message from the queue.
- It calculates the new message.
- Then the distance between the old and new message is calculated.
- This distance is used as the priority with which to add the adjacent messages to the queue.

Parallelisation of RBU

- Memory write clashes are what make this problem difficult.
- The intuitive parallelisation has many memory write clashes.
- For example: Given two threads to do inference on the PGM to the right, the threads will end up waiting for each other to complete work.

Split Message Scheduling

- Split Message Scheduling splits up the message calculation into its two parts: marginalising the source cluster; absorbing the message into the destination cluster.
- This allows threads to spend less time waiting on each other.
- The current best technique has a speedup of 5.2X on satellite image denoising problem using 8 hardware threads.